Runtime randomization and perturbation for virtual machines.

JAVIER CABRERA ARTEAGA

Licentiate Thesis in [Research Subject - as it is in your ISP]
School of Information and Communication Technology
KTH Royal Institute of Technology
Stockholm, Sweden [2022]

KTH School of Information and
Communication Technology
SE-164 40 Kista
SWEDEN

TRITA-ICT XXXX:XX
ISBN XXX-XX-XXXX-XXX-X

Akademisk avhandling som med tillstånd av Kungl Tekniska högskolan framläges till offentlig granskning för avläggande av licentiatexamen i [ämne/subject] [veckodag/weekday] den [dag/day] [månad/month] [år/2022] klockan [tid/time] i [sal/hall], Electrum, Kungl Tekniska högskolan, Kistagången 16, Kista.

© Javier Cabrera Arteaga, [month] [2022]

Tryck: Universitetsservice US AB

Abstract

Write your abstract here...
Keywords: Keyword1, keyword2, ...

Sammanfattning

Write your Swedish summary (popular description) here...
Keywords: Keyword1, keyword2, ...

Acknowledgements

Write your professional acknowledgements here...

Acknowledgements are used to thank all persons who have helped in carrying out the research and to the research organizations/institutions and/or companies for funding the research.

Name Surname,
Place, Date

Contents

Contents **vi**

List of Figures **vii**

List of Tables **viii**

List of Acronyms **ix**

1 Introduction **1**
1.1 Motivation 1
1.1.1 Why variants ? 1
1.2 Contributions 1

2 Background and State of the art **3**
2.1 CROW 3

3 Methodology **5**
3.1 Corpora 5
3.2 RQ1. To what extent we can generate program variants for WebAssembly? 6
3.2.1 Metrics 7
3.2.2 Protocol 8
3.3 RQ2. To what extent the artifically generated variants are different? 8
3.3.1 Metrics 9
3.3.2 Protocol 11
3.4 RQ3. To what extent the artificially generated variants can be used to enforce security on Edge-Cloud platforms? 11
3.4.1 Metrics 12
3.4.2 Protocol 13
3.5 Conclusions 14

Appended papers **14**

List of Figures

2.1 CROW workflow to generate program variants. CROW takes C/C++ source codes or LLVM bitcodes to look for code blocks that can be replaced by semantically equivalent code and generates program variants by combining them. 3

3.1 Simplification of the program variants generation workflow. 7
3.2 Population study methodology for each original corpora and the variants generated in RQ1. 9
3.3 Multivariant binary creation and workflow for RQ3. 12

List of Tables

3.1 Corpora description. The table is composed by the name of the corpus, the number of programs, the number of functions, the lines of code range and the location of the corpus. 7

List of Acronyms

Wasm	WebAssembly
DTW	Dynamic Time Warping